

APELLIDOS	GRUPO DE CLASE:	Firma
NOMBRE		
DNI:		

PROBLEMA 1A (10 puntos)

La siguiente entidad y arquitectura corresponden a un circuito combinacional.

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY combinacional1 IS
PORT (a,b: IN STD_LOGIC;
      sal: OUT STD_LOGIC);
END combinacional1;

ARCHITECTURE estructural OF combinacional1 IS

    COMPONENT inv IS GENERIC (retardo:TIME:= 0 ns);
    PORT(e:IN STD_LOGIC; s:OUT STD_LOGIC);
    END COMPONENT;

    COMPONENT and2 IS GENERIC (retardo:TIME:= 0 ns);
    PORT(e0,e1:IN STD_LOGIC; s:OUT STD_LOGIC);
    END COMPONENT;

    COMPONENT nor2 IS GENERIC (retardo:TIME:= 0 ns);
    PORT(e0,e1:IN STD_LOGIC; s:OUT STD_LOGIC);
    END COMPONENT;

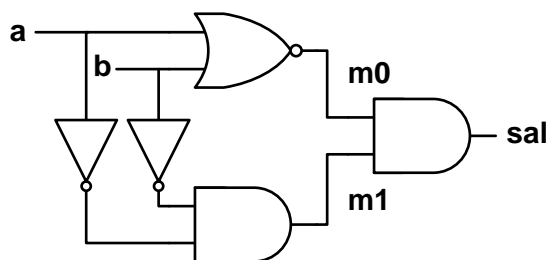
    SIGNAL na,nb,m0,m1:STD_LOGIC;

    FOR ALL: inv USE ENTITY work.inv(flujo);
    FOR ALL: and2 USE ENTITY work.and2(flujo);
    FOR ALL: nor2 USE ENTITY work.nor2(flujo);

BEGIN
    u0: inv      GENERIC MAP (1 ns) PORT MAP (e=>b,      s=>nb);
    u1: inv      GENERIC MAP (1 ns) PORT MAP (e=>a,      s=>na);
    u2: and2     GENERIC MAP (2 ns) PORT MAP (e0=>na,    e1=>nb,      s=>m1);
    u3: nor2     GENERIC MAP (2 ns) PORT MAP (e0=>a,    e1=>b,      s=>m0);
    u4: and2     GENERIC MAP (3 ns) PORT MAP (e0=>m0,   e1=>m1,      s=>sal);
END estructural;

```

1) Dibuje la estructura del circuito.



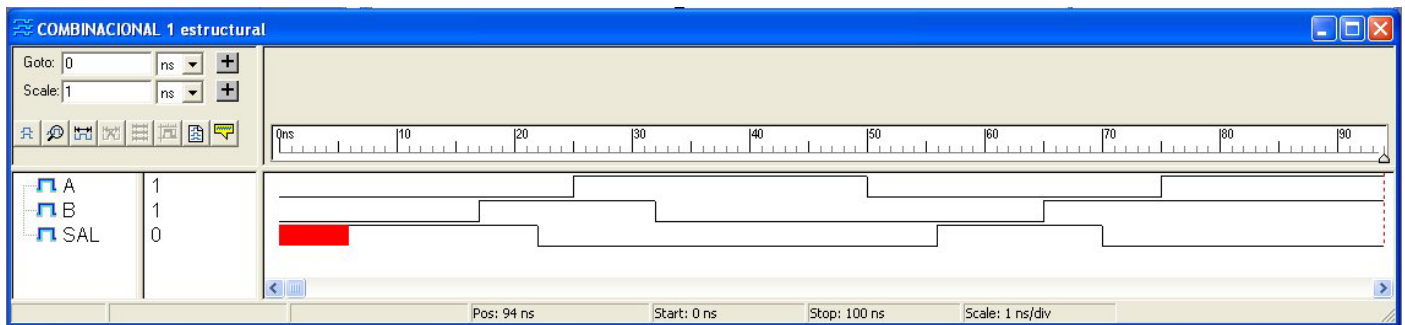
2) Dígase, de forma razonada, con cuantas sentencias concurrentes se ha descrito la arquitectura.

hay 5: u0, u1, u2, u3, u4

3) Dígase, de forma razonada, cuanto sería el retardo máximo de la salida respecto de las entradas.

6 ns = inversor (1ns) + and2 o nor2 (2 ns) and2 + and2 (3ns) (3 niveles de puertas)

4) Si se deseara testear la arquitectura con las formas de onda (A, B) de la figura siguiente, descríbalas con sentencias VHDL.



A <= NOT A AFTER 25 ns;

B <= '0', '1' AFTER 17 ns, '0' AFTER 32 ns, '1' AFTER 65 ns;

5) Obtenga la función de salida generada (sal) y justifique razonadamente si corresponde a la señal (SAL) representada en el cronograma del apartado 4.

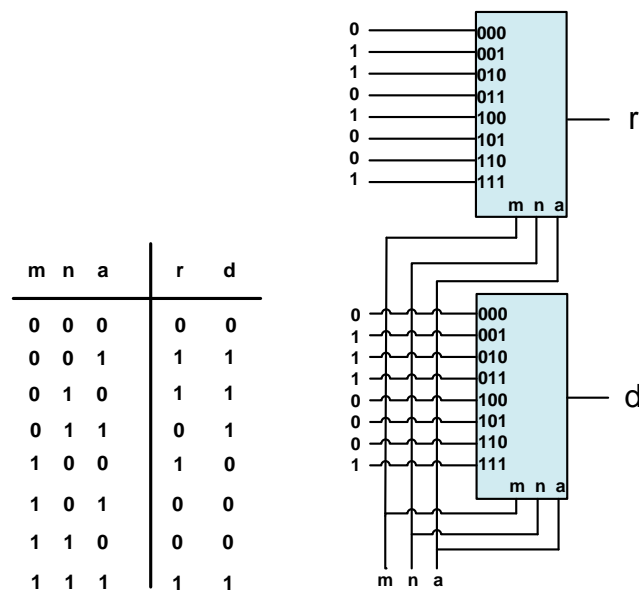
$$Sal = (a+b)' \cdot (a' \cdot b') = a' \cdot b'$$

En la señal SAL del cronograma durante los 6 primeros nanosegundos se obtiene un valor indeterminado, porque no se ha inicializado la salida y como consecuencia del retardo acumulado por el funcionamiento de los 3 niveles de puertas. Después la salida (SAL) responde a la función lógica obtenida en **sal**.

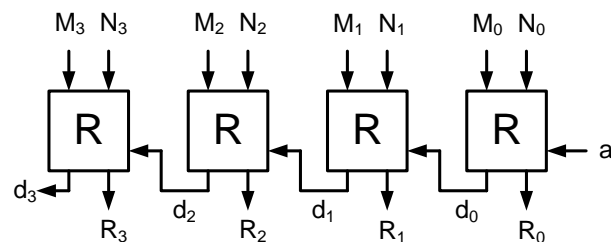
PROBLEMA 2A (10 puntos)

Se desea diseñar un circuito capaz de restar ($M-N$) dos palabras de cuatro bits $M=M_3M_2M_1M_0$ y $N=N_3N_2N_1N_0$ codificados en binario natural, de manera análoga al diseño de un sumador binario; para ello:

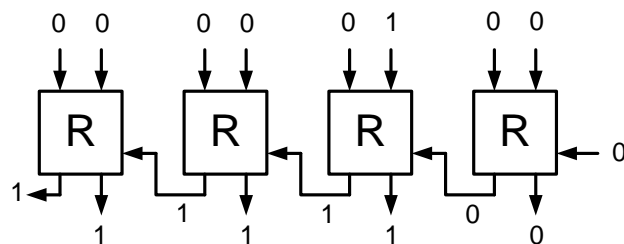
- 1) Escribir la tabla lógica de un restador completo con tres entradas: m , n , y el adeudo a , de una posible etapa anterior y dos salidas: el bit de resta r y el adeudo d a la etapa siguiente. Implementarlo con multiplexores de tres entradas de control.



- 2) Construir el restador de cuatro bits en base a los restadores completos anteriores.



- 3) Supóngase que las entradas (expresadas en decimal) son $M=0$ y $N=2$. ¿Cuál sería el resultado del anterior circuito?. ¿Sería correcto si se interpretase como un número en complemento a dos?



Resultado: 1110

El resultado interpretado como un número en complemento a dos sería -2

- 4) Escribase dicho número, interpretado como un número en complemento a dos, para que esté correctamente representado en 8 bits. ¿Cuál es el rango de representación de los números binarios de cuatro bits y el de los números en complemento a dos de cuatro bits?

En ocho bits $\rightarrow 11111110$

Rango de números binarios de cuatro bits $[0, \dots, 15]$

Rango de números en complemento a dos de cuatro bits $[-8, \dots, 0, \dots, +7]$

PROBLEMA 3A (10 puntos)

Dados dos números menores que 4: A (bits a_i siendo $i=n-1, \dots, 0$) y B (bits b_i siendo $i=n-1, \dots, 0$) codificados en binario natural, se quiere diseñar un circuito cuya salida S (bits s_j siendo $j=m-1, \dots, 0$), sean los números binarios de entrada ordenados de mayor a menor. Se pide:

- 1) Número de entradas y salidas del circuito. (1 punto)
- 2) Obtención de la tabla de verdad, de acuerdo a las siguientes cabeceras. (3 puntos)

$a_{n-1} \dots a_0$	$b_{n-1} \dots b_0$	$S_{m-1} \dots S_0$
---------------------	---------------------	---------------------

- 3) Reducción de las funciones lógicas de las dos salidas de mayor peso mediante mapas de Karnaugh. (2 puntos)
- 4) De las dos funciones lógicas obtenidas en el apartado anterior, implementación de la de menor peso con puertas NAND. (2 puntos)
- 5) Implementación de la salida de menor peso del circuito con un multiplexor, controlado por los bits del número A. (2 puntos)

1) Entradas = 4 ; Salidas = 4

2)

A_2		A_{10}	B_2		B_{10}	S_3	S_2	$V_{S_3S_2}$	S_1	S_0	$V_{S_1S_0}$
a_1	a_0		b_1	b_0							
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	0	1	1	0	0	0
0	0	0	1	0	2	1	0	2	0	0	0
0	0	0	1	1	3	1	1	3	0	0	0
0	1	1	0	0	0	0	1	1	0	0	0
0	1	1	0	1	1	0	1	1	0	1	1
0	1	1	1	0	2	1	0	2	0	1	1
0	1	1	1	1	3	1	1	3	0	1	1
1	0	2	0	0	0	1	0	2	0	0	0
1	0	2	0	1	1	1	0	2	0	1	1
1	0	2	1	0	2	1	0	2	1	0	2
1	0	2	1	1	3	1	1	3	1	0	2
1	1	3	0	0	0	1	1	3	0	0	0
1	1	3	0	1	1	1	1	3	0	1	1
1	1	3	1	0	2	1	1	3	1	0	2
1	1	3	1	1	3	1	1	3	1	1	3

3)

a_1a_0				
b_1b_0	00	01	11	10
00			1	1
01			1	1
11	1	1	1	1
10	1	1	1	1

$$S_3 = b_1 + a_1$$

a_1a_0				
b_1b_0	00	01	11	10
00		1	1	
01	1	1	1	
11	1	1	1	1
10			1	

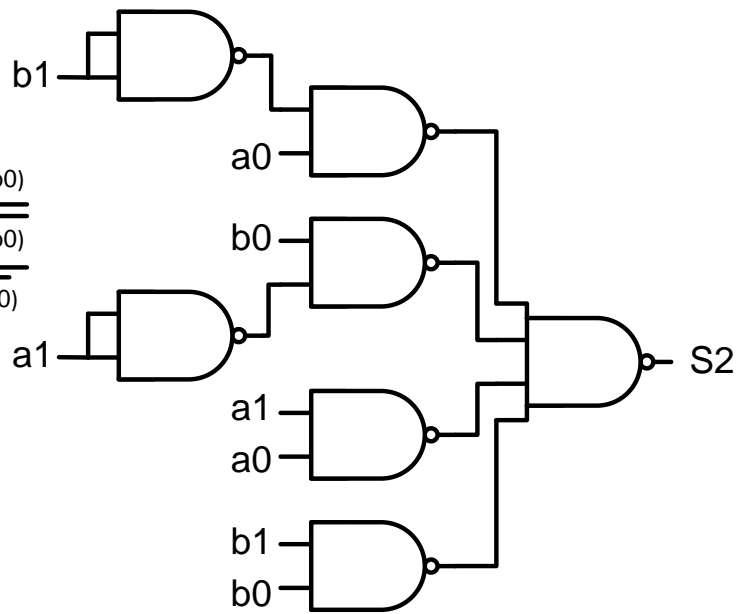
$$S_2 = (a_0 \cdot b_1') + (a_1' \cdot b_0) + (a_1 \cdot a_0) + (b_1 \cdot b_0)$$

4)

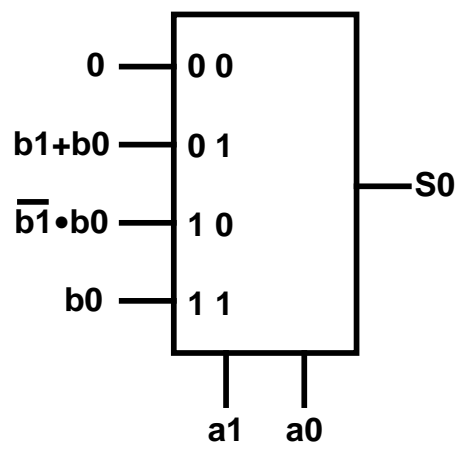
$$S2 = \overline{b1} \cdot a0 + b0 \cdot \overline{a1} + a1 \cdot a0 + b1 \cdot b0$$

$$S2 = \overline{b1 \cdot a0} + \overline{b0 \cdot a1} + \overline{a1 \cdot a0} + \overline{b1 \cdot b0}$$

$$S2 = \overline{b1 \cdot a0} \cdot \overline{b0 \cdot a1} \cdot \overline{a1 \cdot a0} \cdot \overline{b1 \cdot b0}$$



5)



PROBLEMA 1B (10 puntos)

La siguiente entidad y arquitectura corresponden a un circuito combinacional.

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY combinacional2 IS
PORT (a,b: IN STD_LOGIC;
      sal: OUT STD_LOGIC);
END combinacional2;

ARCHITECTURE estructural OF combinacional2 IS

    COMPONENT inv IS GENERIC (retardo:TIME:= 0 ns);
    PORT(e:IN STD_LOGIC; s:OUT STD_LOGIC);
    END COMPONENT;

    COMPONENT and2 IS GENERIC (retardo:TIME:= 0 ns);
    PORT(e0,e1:IN STD_LOGIC; s:OUT STD_LOGIC);
    END COMPONENT;

    COMPONENT or2 IS GENERIC (retardo:TIME:= 0 ns);
    PORT(e0,e1:IN STD_LOGIC; s:OUT STD_LOGIC);
    END COMPONENT;

    COMPONENT nand2 IS GENERIC (retardo:TIME:= 0 ns);
    PORT(e0,e1:IN STD_LOGIC; s:OUT STD_LOGIC);
    END COMPONENT;

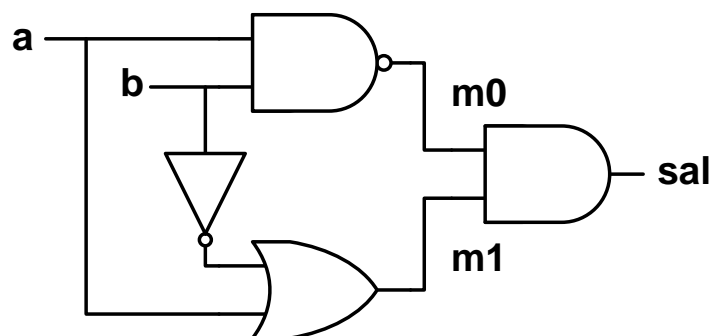
    SIGNAL nb,m0,m1:STD_LOGIC;

    FOR ALL: inv USE ENTITY work.inv(flujo);
    FOR ALL: and2 USE ENTITY work.and2(flujo);
    FOR ALL: or2 USE ENTITY work.or2(flujo);
    FOR ALL: nand2 USE ENTITY work.nand2(flujo);

BEGIN
    u0: inv      GENERIC MAP (1 ns) PORT MAP (e=>b,          s=>nb);
    u1: or2      GENERIC MAP (2 ns) PORT MAP (e0=>a,          e1=>nb,          s=>m1);
    u2: nand2    GENERIC MAP (2 ns) PORT MAP (e0=>a,          e1=>b,           s=>m0);
    u3: and2     GENERIC MAP (3 ns) PORT MAP (e0=>m0,         e1=>m1,          s=>sal);
END estructural;

```

1) Dibuje la estructura del circuito.



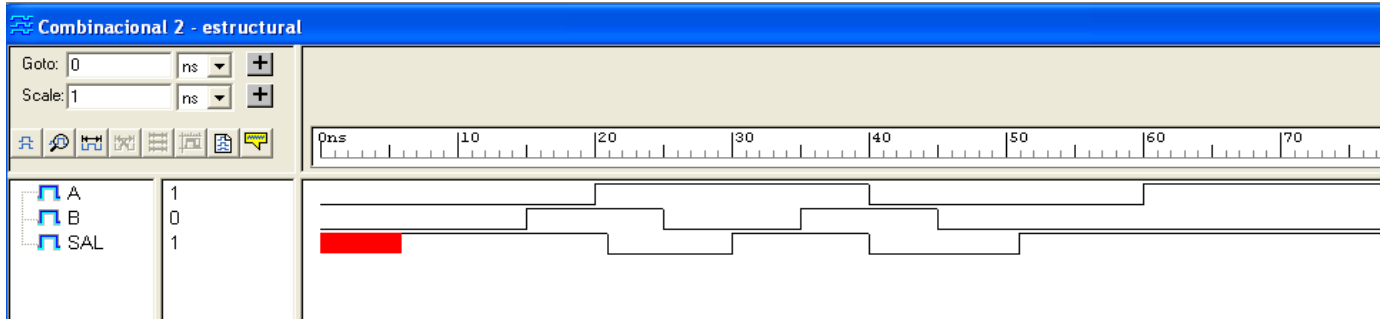
2) Dígase, de forma razonada, con cuantas sentencias concurrentes se ha descrito la arquitectura.

hay 4: u0, u1, u2, u3

3) Dígase, de forma razonada, cuanto sería el retardo máximo de la salida respecto de las entradas.

son 6 ns = inversor (1 ns) + or2 o nand2 (2 ns) + and2 (3 ns) (3 niveles de puertas)

4) Si se deseara testear la arquitectura con las formas de onda (A, B) de la figura siguiente, descríbalas con sentencias VHDL.



A <= NOT A AFTER 20 ns;

B <= '0', '1' AFTER 15 ns, '0' AFTER 25 ns, '1' AFTER 35 ns, '0' AFTER 45 ns;

5) Obtenga la función de salida generada (sal) y justifique razonadamente si corresponde a la señal (SAL) representada en el cronograma del apartado 4.

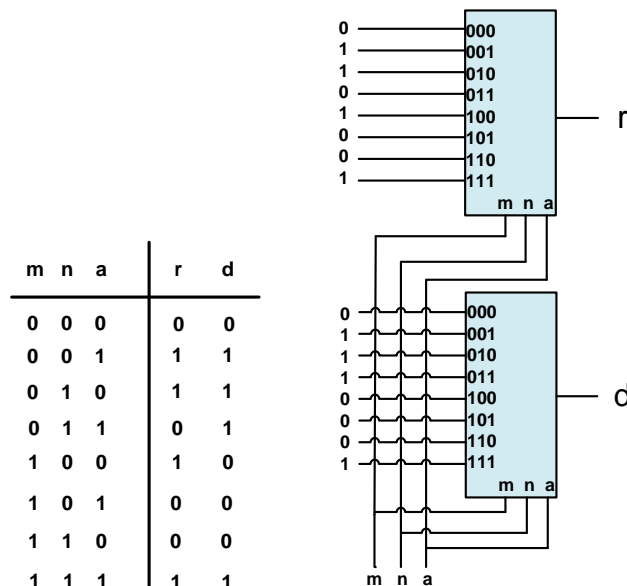
$$\text{Sal} = (a \cdot b)' \cdot (a + b') = (a' + b') \cdot (a + b') = b'$$

En la señal SAL del cronograma durante los 6 primeros nanosegundos se obtiene un valor indeterminado, porque no se ha inicializado la salida y como consecuencia del retardo acumulado por el funcionamiento de los 3 niveles de puertas. Después la salida (SAL) responde a la función lógica obtenida en **sal**.

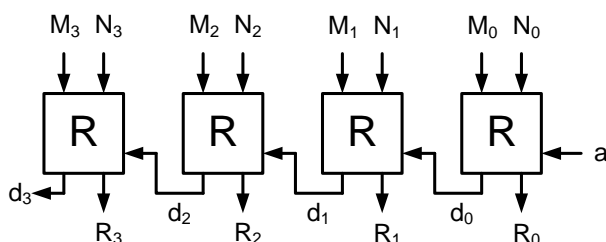
PROBLEMA 2B (10 puntos)

Se desea diseñar un circuito capaz de restar ($M-N$) dos palabras de cuatro bits $M=M_3M_2M_1M_0$ y $N=N_3N_2N_1N_0$ codificados en binario natural, de manera análoga al diseño de un sumador binario; para ello:

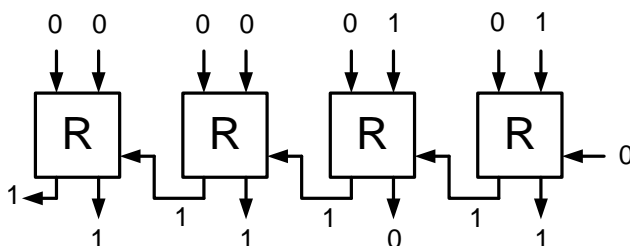
1. Escribir la tabla lógica de un restador completo con tres entradas: m , n , y el adeudo a , de una posible etapa anterior y dos salidas: el bit de resta r y el adeudo d a la etapa siguiente. Implementarlo con multiplexores de tres entradas de control.



2. Construir el restador de cuatro bits en base a los restadores completos anteriores.



3. Supóngase que las entradas (expresadas en decimal) son $M=0$ y $N=3$. ¿Cuál sería el resultado del anterior circuito?. ¿Sería correcto si se interpretase como un número en complemento a dos?



Resultado: 1101

El resultado interpretado como un número en complemento a dos sería -3

4. Escribese dicho número, interpretado como un número en complemento a dos, para que esté correctamente representado en 10 bits. ¿Cuál es el rango de representación de los números binarios de diez bits y el de los números en complemento a dos de diez bits?

En ocho bits \rightarrow 11111101

Rango de números binarios de diez bits $[0, \dots, 1023]$

Rango de números en complemento a dos de diez bits $[-512, \dots, 0, \dots, +511]$

PROBLEMA 3B (10 puntos)

Dados dos números menores que 4: A (bits a_i siendo $i=n-1, \dots, 0$) y B (bits b_i siendo $i=n-1, \dots, 0$) codificados en binario natural, se quiere diseñar un circuito cuya salida S (bits s_j siendo $j=m-1, \dots, 0$), sean los números binarios de entrada ordenados de mayor a menor. Se pide:

- 1) Número de entradas y salidas del circuito. (1 punto)
- 2) Obtención de la tabla de verdad, de acuerdo a las siguientes cabeceras. (3 puntos)

$a_{n-1} \dots a_0$	$b_{n-1} \dots b_0$	$S_{m-1} \dots S_0$
---------------------	---------------------	---------------------

- 3) Reducción de las funciones lógicas de las dos salidas de menor peso mediante mapas de Karnaugh. (2 puntos)
- 4) De las dos funciones lógicas obtenidas en el apartado anterior, implementación de la de menor peso con puertas NAND. (2 puntos)
- 5) Implementación de la salida de mayor peso del circuito con un multiplexor, controlado por los bits del número A. (2 puntos)

1) Entradas = 4 ; Salidas = 4

2)

A_2		A_{10}	B_2		B_{10}	S_3	S_2	$V_{S_3S_2}$	S_1	S_0	$V_{S_1S_0}$
a_1	a_0		b_1	b_0							
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	0	1	1	0	0	0
0	0	0	1	0	2	1	0	2	0	0	0
0	0	0	1	1	3	1	1	3	0	0	0
0	1	1	0	0	0	0	1	1	0	0	0
0	1	1	0	1	1	0	1	1	0	1	1
0	1	1	1	0	2	1	0	2	0	1	1
0	1	1	1	1	3	1	1	3	0	1	1
1	0	2	0	0	0	1	0	2	0	0	0
1	0	2	0	1	1	1	0	2	0	1	1
1	0	2	1	0	2	1	0	2	1	0	2
1	0	2	1	1	3	1	1	3	1	0	2
1	1	3	0	0	0	1	1	3	0	0	0
1	1	3	0	1	1	1	1	3	0	1	1
1	1	3	1	0	2	1	1	3	1	0	2
1	1	3	1	1	3	1	1	3	1	1	3

3)

a_1a_0	00	01	11	10
b_1b_0				
00				
01				
11			1	1
10			1	1

$$S_1 = a_1 \cdot b_1$$

a_1a_0	00	01	11	10
b_1b_0				
00				
01		1	1	1
11		1	1	
10		1		

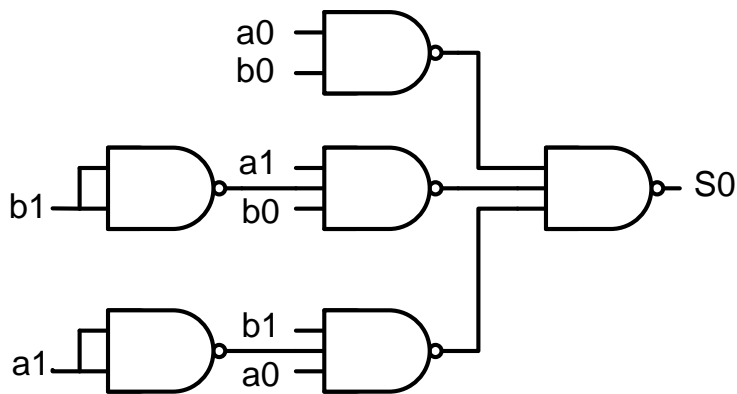
$$S_0 = (a_0 \cdot b_0) + (a_1 \cdot b_1' \cdot b_0) + (b_1 \cdot a_1' \cdot a_0)$$

4)

$$S0 = (a0 \ b0) + (a1 \ \overline{b1} \ b0) + (b1 \ \overline{a1} \ a0)$$

$$S0 = (a0 \ b0) + (a1 \ \overline{b1} \ b0) + (b1 \ \overline{a1} \ a0)$$

$$S0 = (a0 \ b0) \cdot (a1 \ \overline{b1} \ b0) \cdot (b1 \ \overline{a1} \ a0)$$



5)

